

AX8905

**Intelligent System Monitoring
Module for True Remote
Management**

User's Manual

Disclaimers

The information in this manual has been carefully checked and is believed to be accurate. AXIOMTEK Co., Ltd. assumes no responsibility for any infringements of patents or other rights of third parties which may result from its use.

AXIOMTEK assumes no responsibility for any inaccuracies that may be contained in this document. AXIOMTEK makes no commitment to update or to keep current the information contained in this manual.

AXIOMTEK reserves the right to make improvements to this document and/or product at any time and without notice.

No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of AXIOMTEK Co., Ltd.

©Copyright 2002 by AXIOMTEK Co., Ltd.
All rights reserved.
November 2002, Version A2
Printed in Taiwan

ESD Precautions

Integrated circuits on computer boards are sensitive to static electricity. To avoid damaging chips from electrostatic discharge, observe the following precautions:

- Do not remove boards or integrated circuits from their anti-static packaging until you are ready to install them.
- Before handling a board or integrated circuit, touch an unpainted portion of the system unit chassis for a few seconds. This helps to discharge any static electricity on your body.
- Wear a wrist-grounding strap, available from most electronic component stores, when handling boards and components.

Trademarks Acknowledgments

AXIOMTEK is a trademark of AXIOMTEK Co., Ltd.

Award is a registered trademark of Award Software International, Inc.

IBM, PS/2 are trademarks of International Business Machines Corporation.

Intel and Pentium are registered trademarks of Intel Corporation.

Microsoft Windows is a registered trademark of Microsoft Corporation.

Other brand names and trademarks are the properties and registered brands of their respective owners.

This page does not contain any information.

Table of Contents

Chapter 1 Introduction-----	1
1.1 General Description	1
1.2 Utility Architecture (for Windows)	2
1.3 Utility Architecture (for Linux).....	2
1.4 Features	3
1.5 Location Diagram	4
Chapter 2 Utility Installation -----	9
2.1 Windows	9
2.2 Linux	9
Chapter 3 GUI Monitoring Utility -----	11
Chapter 4 About Homepage -----	19
4.1 Default.htm	19
4.2 Default.asp.....	23
4.3 Index.html and AX8905.cgi	25
Chapter 5 Function List -----	27
5.1 Windows	27
5.1.1 ConnectRemoteHost.....	27
5.1.2 GetVoltEnable	27
5.1.3 GetVoltAlarm	28
5.1.4 GetVoltSetting	28
5.1.5 GetVoltReading	29
5.1.6 GetTempEnable	29
5.1.7 GetTempAlarm	30
5.1.8 GetTempSetting	30
5.1.9 GetTempReading	31
5.1.10 GetFanEnable.....	31
5.1.11 GetFanAlarm.....	32
5.1.12 GetFanSetting	32
5.1.13 GetFanReading.....	33
5.1.14 GetDigitalEnable.....	33
5.1.15 GetDigitalAlarm.....	34
5.1.16 GetTrimFanSpeed	34
5.1.17 GetTelNo	35
5.1.18 SetVoltEnable.....	35
5.1.19 SetVoltSetting.....	36

5.1.20	SetTempEnable	36
5.1.21	SetTempSetting	37
5.1.22	SetFanEnable	37
5.1.23	SetFanSetting	38
5.1.24	SetDigitalEnable	38
5.1.25	SetTrimFanSpeed	39
5.1.26	SetTelNo.....	39
5.1.27	SendAllValueToRemoteHost.....	40
5.2	Linux	40
5.2.1	GetRemoteHostData.....	40
5.2.2	GetVoltEnable	40
5.2.3	GetVoltAlarm	41
5.2.4	GetVoltSetting	41
5.2.5	GetVoltReading	42
5.2.6	GetTempEnable	42
5.2.7	GetTempAlarm	43
5.2.8	GetTempSetting	43
5.2.9	GetTempReading	44
5.2.10	GetFanEnable.....	44
5.2.11	GetFanAlarm.....	45
5.2.12	GetFanSetting	45
5.2.13	GetFanReading.....	46
5.2.14	GetDigitalEnable.....	46
5.2.15	GetDigitalAlarm.....	47
5.2.16	GetTrimFanSpeed	47
5.2.17	GetTelNo	48
5.2.18	SetVoltEnable.....	48
5.2.19	SetVoltSetting.....	48
5.2.20	SetTempEnable.....	49
5.2.21	SetTempSetting.....	49
5.2.22	SetFanEnable	50
5.2.23	SetFanSetting	50
5.2.24	SetDigitalEnable	51
5.2.25	SetTrimFanSpeed	51
5.2.26	SetTelNo.....	52
5.2.27	SendDataToRemoteHost	52

Chapter 1

Introduction

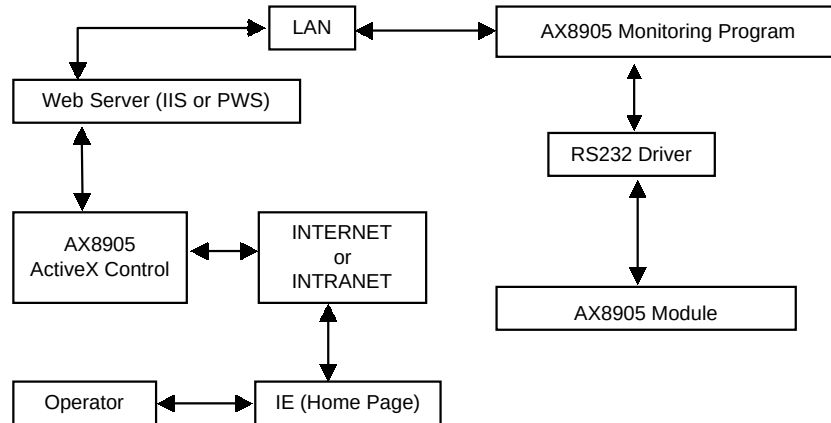
1.1 General Description

AX8905 provides an ultimate solution to monitor the various critical signs of an industrial computer system. With the pre-installed GUI software, this new module offers Information Technology (IT) administrators extensive capabilities to manage the systems through Web.

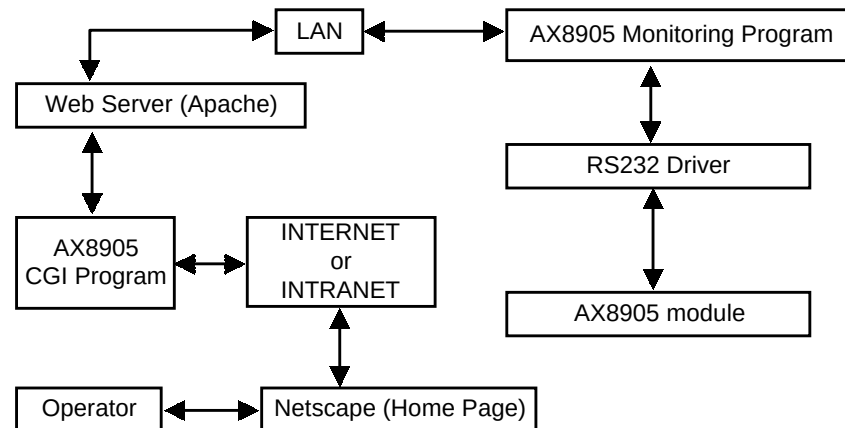
The AX8905 regulates fan speed, internal temperature of the system, and the power voltage in a real-time base, providing audible alarm as well as LED indication. It allows IT administrators define specific groups of systems within a network so to monitor an entire system's performance at a glance. Once the system is running abnormally, the IT administrators can immediately detect bottlenecks and analyze critical points using the Web-based interface. The warning can be transmitted through paging notification, and direct phone line, so the IA administrators now can go anywhere with the system status report on their fingertips.

Initially, the AX8905 is configured in the AX6115 and AX6125 Internet Servers as a system health-monitoring module. This has made AXIOMTEK's new Internet Servers outstanding from the crowd and become more preferable to the environments and requirements of ISP servers. Except for such applications, the AX8905 can also be integrated into other industrial computer system, in which critical mission is a must.

1.2 Utility Architecture (for Windows)



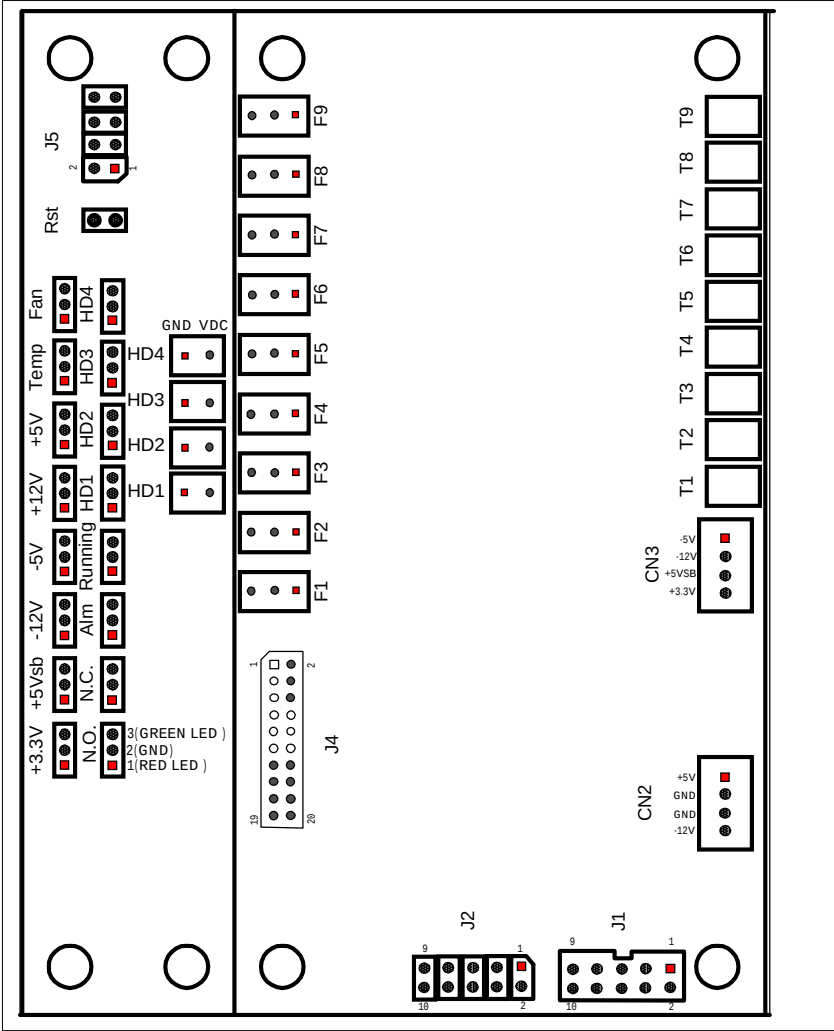
1.3 Utility Architecture (for Linux)



1.4 Features

- Temperature limit detection x 9
- Voltage limit detection x 6 (+3.3V, +5V, +5Vsb, +12V, -5V, -12V)
- Fan speed detection x 9
- Fan rotation speed control
- Digital input detection x 2 (Normal Open ; Normal Close)
- Modem dial up when the alarm occurs
- LED indication
- PC user interface for setting limit value and reading detected value
- Transmission of information through Web
- Alarm Buzzer
- Pre-installed GUI software makes the linkage of two ends between client and server via Internet. The performance of monitoring task is as good as the local operation with real-time responses.
- Windows 95/98/NT and Linux (RedHat 6.x) utility provided

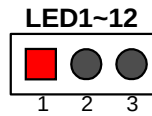
1.5 Location Diagram



Descriptions:

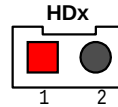
- **+3.3V, +5Vsb, -12V, -5V, +12V, +5V, Temp (temperature), Fan, N.C.(Normal closed) and N.O.(Normal opened) LEDs** : For the corresponding alarm condition. When alarm condition occurs, the LED will change to RED color. It will be GREEN when normal condition.
- **Alm LED**: Indicate alarm enable or disable.
- **Running LED**: When blinking, it means the monitoring board is working.

Pin #	Signal Name
1	Red LED Power
2	LED Ground
3	Green LED power



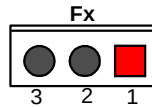
- **HDx LEDs**: The status of HD1, HD2, HD3 and HD4.
- **HDx** : Connector of HD1, HD2, HD3 and HD4.

Pin #	Signal Name
1	Ground
2	HD +VDC



- **Fx**: Fan connector.

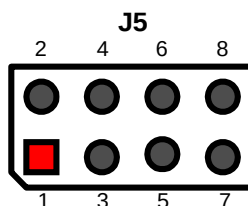
Pin #	Signal Name
1	Ground
2	Power
3	Fan Signal



- **Tx**: Temperature sensor connector.

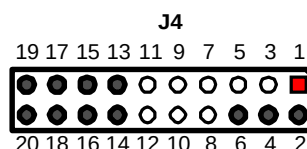
■ **J5: Connect to J4**

Pin #	Connect to J4 Pin #
1	Pin 13
2	Pin 14
3	Pin 15
4	Pin 16
5	Pin 17
6	Pin 18
7	Pin 19
8	Pin 20



■ **J4: 20Pin Pin header**

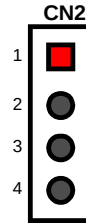
Pin #	Signal Name
2	N.C.
4	GND
6	N.O.



- **N.C.:** TTL level digital input alarm. The normal status is closed to GND.
- **N.O.:** TTL level digital input alarm. The normal status is opened from GND.
- **Rst :** Connector of alarm reset button. When power on (default), the alarm function is enabled. At the same time, you can notice that Alm LED is green. When alarm happens, the alarm function will be disabled automatically for next alarm trigger. At this moment, you can see that Alm LED turns to red. To enable alarm function again, push Rst button until Alm LED turns to green. By doing so, the previous alarm is cleared. And the next alarm trigger is allowed to come in now. If you want to disable alarm function manually, push Rst button again until Alm LED turns to red.

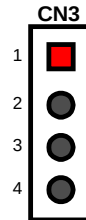
■ **CN2:** Power supply connector.

Pin #	Signal Name
1	+5VDC
2	Ground
3	Ground
4	-12VDC



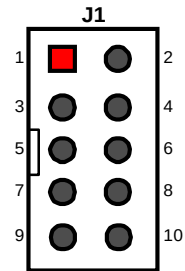
■ **CN3:** Input voltage for monitoring. There are +3.3V, +5Vsb, -12V, -5V. CN3 pin assignment is as follows:

Pin #	Signal Name
1	-5VDC
2	-12VDC
3	+5VSB
4	+3.3VDC



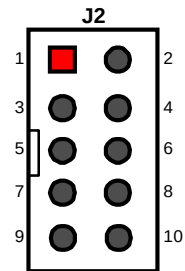
■ **J1:** Connect to COM port of PC.

Pin #	Signal Name	Pin #	Signal Name
1	NC	2	NC
3	TXD	4	NC
5	RXD	6	NC
7	NC	8	NC
9	GND	10	NC



■ **J2:** Connect to MODEM for dial-up.

Pin #	Signal Name	Pin #	Signal Name
1	DCD	2	DSR
3	RXD	4	RTS
5	TXD	6	CTS
7	DTR	8	RI
9	GND	10	NC



This page does not contain any information.

Chapter 2

Utility Installation

The utility installation procedures are described on the following sections of this chapter.

2.1 Windows

Insert the installation disk in your PC's floppy driver and execute "setup.exe" under Windows. The setup utility will copy necessary files to your system.

2.2 Linux

1. Copy AX8905MT (monitoring program) to any path, then you can run this application.
2. Set up Apache in PC.
3. Modify three config files (/etc/httpd/conf/httpd.conf, /etc/httpd/conf/srm.conf, /etc/httpd/conf/access.conf) if you have some special settings. Remember if you modified config files, please restart computer.
4. Copy the AX8905.cgi to /home/httpd/cgi-bin/.
5. Copy the index.html to /home/httpd/html/.
6. Copy two '.gif' files to /home/httpd/icons/.

You may redesign the homepage. Modify AX8905.c and compile it as follows:

```
gcc AX8905.c AX8905L.o -o AX8905.cgi
```

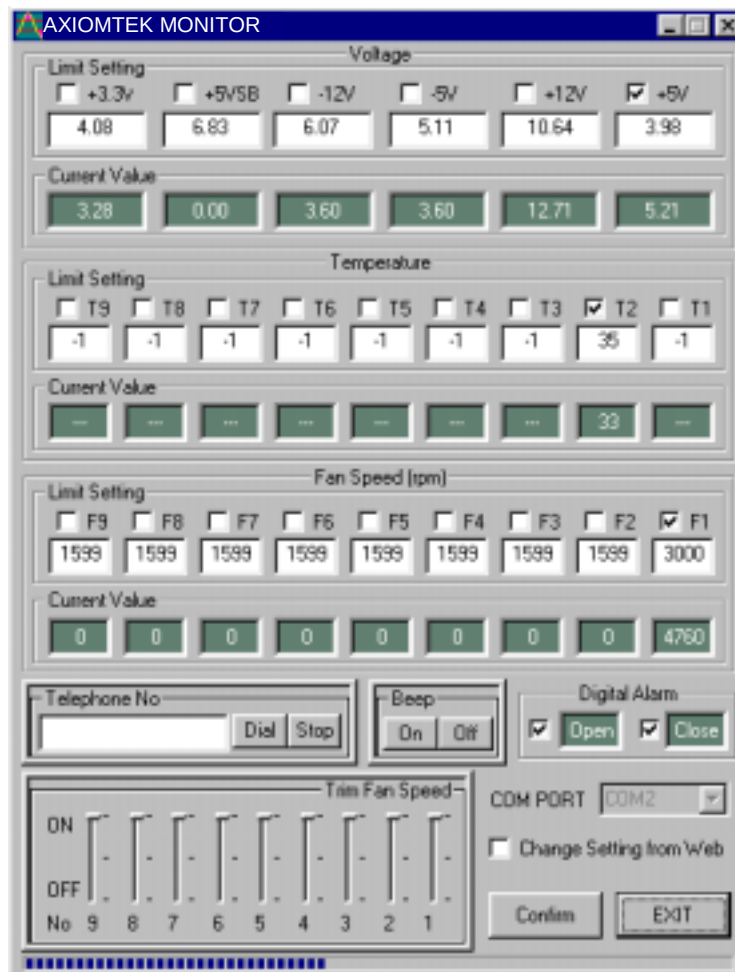
Details of functions provided by AX8905L are listed in Chapter 5.

This page does not contain any information.

Chapter 3

GUI Monitoring Utility

When AXIOMTEK monitoring program starts running, the screen displays a **control panel** on the screen. The control panel in Windows and Linux is similar. The detailed descriptions are on the following sections.



Voltage panel:

- Check boxes can let you disable or enable alarm of monitoring item.
- Edit boxes can let you change individual voltage minimum value for alarming.
- Current value edit boxes show the voltage read back. If alarm occurs, it will turn red

When alarm has arrived, the screen displays as follows:

The screenshot shows the AXIOMTEK MONITOR GUI with three main monitoring panels: Voltage, Temperature, and Fan Speed. Each panel has a 'Limit Setting' section with checkboxes and a 'Current Value' section with input boxes. The Voltage panel shows a red '5.21' in the current value box, indicating an alarm. The Temperature and Fan Speed panels show '---' in their current value boxes. The bottom section includes a 'Telephone No.' field, 'Beep' controls, 'Digital Alarm' controls, a 'COM PORT' dropdown, and a 'Trim Fan Speed' section with sliders and buttons.

Voltage					
Limit Setting	☐ +3.3v	☐ +5V/58	☐ -12V	☐ -5V	☐ +12V
	☒ +5V				
Current Value	4.08	6.83	6.07	5.11	10.64
	3.18	0.00	3.60	3.60	12.71

Temperature								
Limit Setting	☐ T9	☐ T8	☐ T7	☐ T6	☐ T5	☐ T4	☐ T3	☒ T2
	☐ T1							
Current Value	-1	-1	-1	-1	-1	-1	-1	35
	---	---	---	---	---	---	---	---

Fan Speed (rpm)								
Limit Setting	☐ F9	☐ F8	☐ F7	☐ F6	☐ F5	☐ F4	☐ F3	☒ F2
	☐ F1							
Current Value	1599	1599	1599	1599	1599	1599	1599	3000
	0	0	0	0	0	0	0	4800

Telephone No: [] Dial Stop

Beep: On Off

Digital Alarm: ☒ Open ☒ Close

COM PORT: COM2

Change Setting from Web: ☐

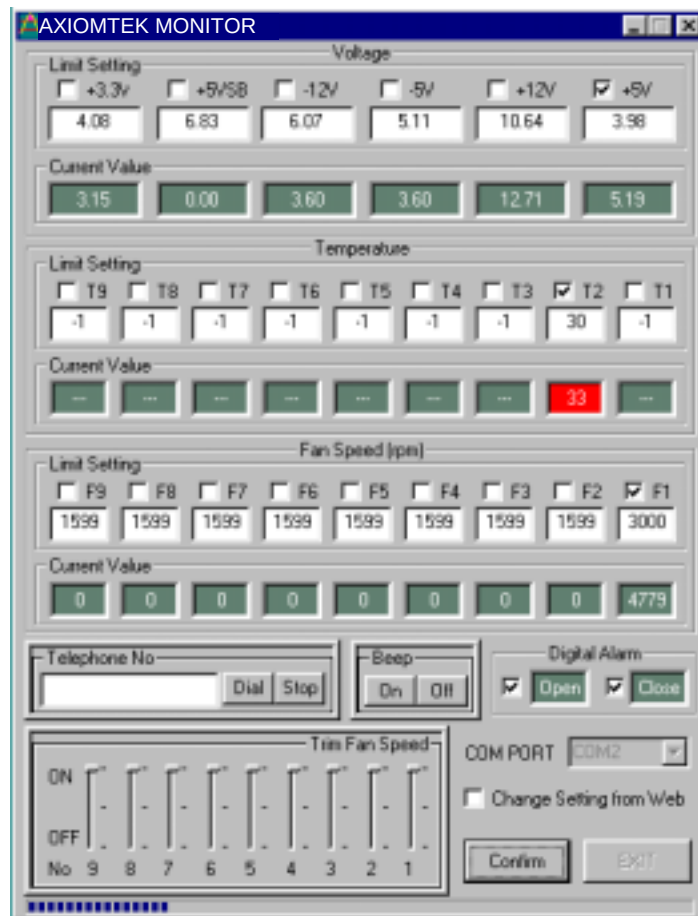
Trim Fan Speed: ON OFF

Confirm Exit

Temperature panel:

- Check boxes can let you disable or enable alarm of monitoring item.
- Edit boxes can let you change individual temperature maximum value.
- Current value edit boxes show the temperature read back. If alarm occurs, it will turn red

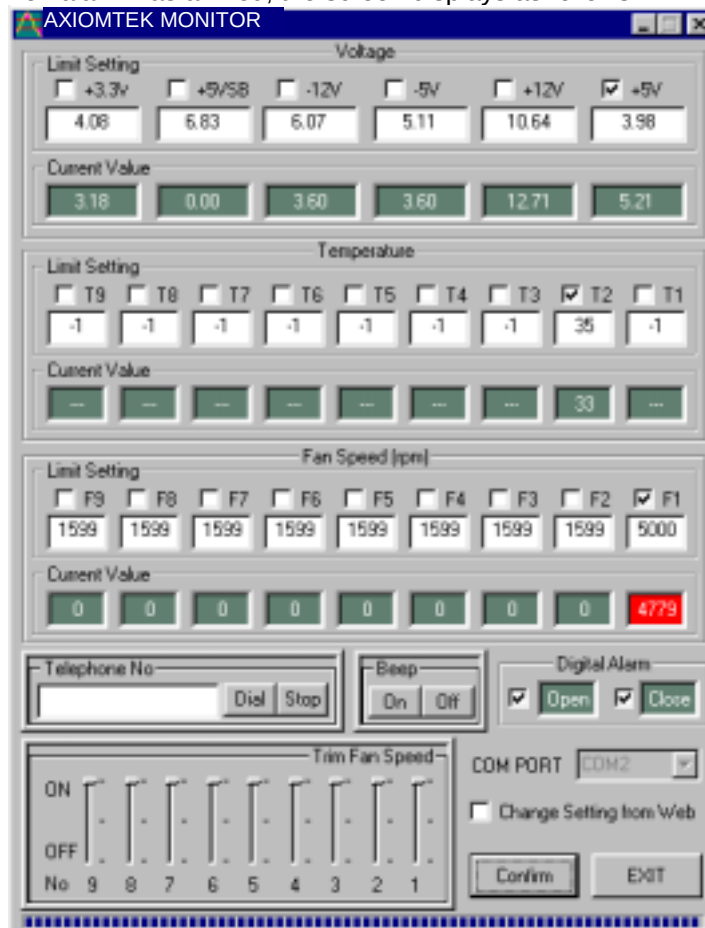
When alarm has arrived, the screen displays as follows:



Fan speed panel:

- Check boxes can let you disable or enable alarm of monitoring item.
- Edit boxes can let you change individual temperature minimum value.
- Current value edit boxes show fan speed read back. If alarm occurs, it will turn red

When alarm has arrived, the screen displays as follows:



Trim fan speed panel:

- Sliders can let you turn individual fan on and off.

The screen displays as follows:

The screenshot displays the AXIOMTEK MONITOR software interface. It features several panels for monitoring and controlling system parameters:

- Voltage Panel:** Includes limit settings for +3.3v, +5VSB, -12V, -5V, +12V, and +5V (checked). Current values are displayed for each.
- Temperature Panel:** Includes limit settings for T1 through T9. T2 is checked with a limit of 35. Current values are displayed for each.
- Fan Speed (rpm) Panel:** Includes limit settings for F1 through F9. F1 is checked with a limit of 3000. Current values are displayed for each.
- Telephone No. Panel:** Includes a text input field and Dial/Stop buttons.
- Beep Panel:** Includes On/Off buttons.
- Digital Alarm Panel:** Includes Open/Close buttons, both of which are checked.
- Trim Fan Speed Panel:** Includes a slider for each of the 9 fans, with labels for ON, OFF, and No. The slider for fan 1 is currently set to OFF.
- COM PORT Panel:** Includes a dropdown menu set to COM2 and a checkbox for Change Setting from Web.
- Buttons:** Confirm and EXIT buttons are located at the bottom right.

Change Setting from web check box:

- Check boxes enable settings to be changed from web.

The screen displays as follows:

AXIOMTEK MONITOR

Voltage

Limit Setting

☐ +3.3v

☐ +5VSB

☐ -12V

☐ -5V

☐ +12V

☒ +5V

4.08

6.83

6.07

5.11

10.64

3.98

Current Value

3.25

0.00

3.60

3.60

12.71

5.21

Temperature

Limit Setting

☐ T9

☐ T8

☐ T7

☐ T6

☐ T5

☐ T4

☐ T3

☒ T2

☐ T1

-1

-1

-1

-1

-1

-1

-1

35

-1

Current Value

33

Fan Speed (rpm)

Limit Setting

☐ F9

☐ F8

☐ F7

☐ F6

☐ F5

☐ F4

☐ F3

☐ F2

☒ F1

1599

1599

1599

1599

1599

1599

1599

1599

3000

Current Value

0

0

0

0

0

0

0

0

4800

Telephone No

Dial

Stop

Beep

On

Off

Digital Alarm

☒ Open

☒ Close

Trim Fan Speed

ON

OFF

No

9

8

7

6

5

4

3

2

1

COM PORT

COM2

☒ Change Setting from Web

Confirm

EXIT

16

GUI Monitoring Utility

COM PORT box:

- It shows the COM Port No. which AX8905 is connected to.

Telephone No. panel:

- Edit boxes let you enter phone/pager no. that will be called when alarm occurs.
- Dial button: click this button and the phone no. in edit box will be dialed out.
- Stop button: click this button to stop dialing out.

Beep panel:

- On button: click this button to manually turn on buzzer. Usually this function is used for buzzer testing.
- Off button: click this button to manually turn off buzzer. Usually this function is used for buzzer testing.

Digital Alarm:

- Open check box: check this box if you want to enable normal-close digital input alarm.
- Close check box: check this box if you want to enable normal-open digital input alarm.

Confirm button:

- It can let you confirm the settings changed.

Exit button:

- When this button is clicked, the application ends.

This page does not contain any information.

Chapter 4

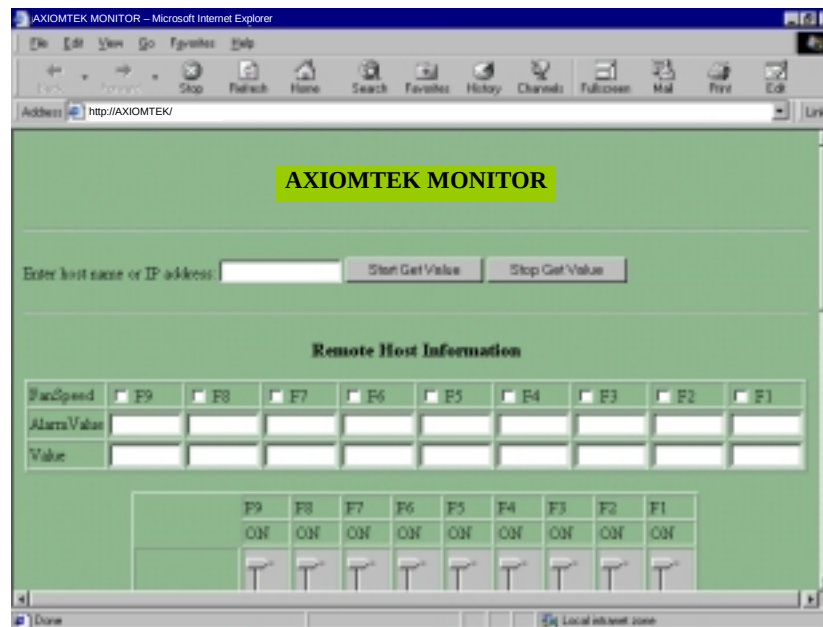
About Homepage

We provide two sets of homepage sample for user reference in Windows environment, that is default.htm and default.asp. In Linux environment, we provide index.html and AX8905.cgi.

4.1 Default.htm

Before starting, you must install and register the AXIOMTEK monitoring ActiveX control. Usually it has been done by setup program. Note that web server is optional. This homepage is just a demo which you how to use HTML and VBScript to control your AX8905.

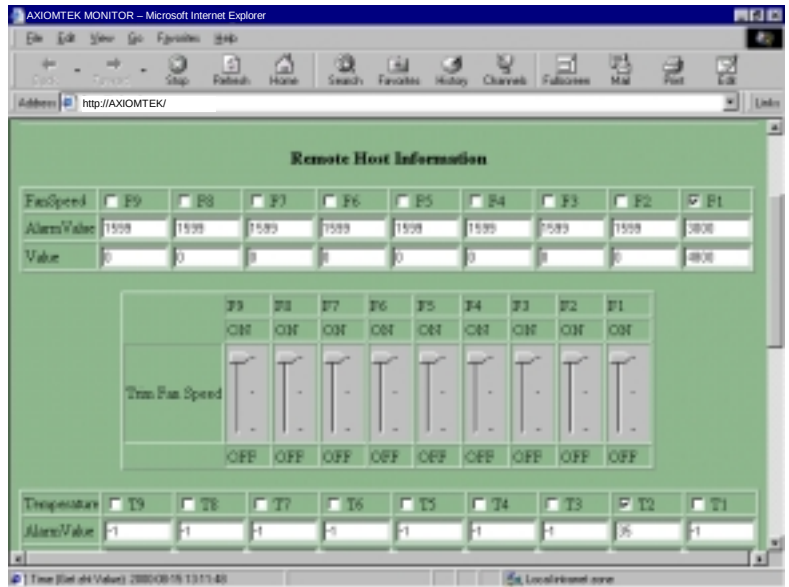
The default.htm is shown as follows:



After entering host name or IP address, click the button named "StartGetValue" and you can get remote host information. The display on screen is shown on the next page.



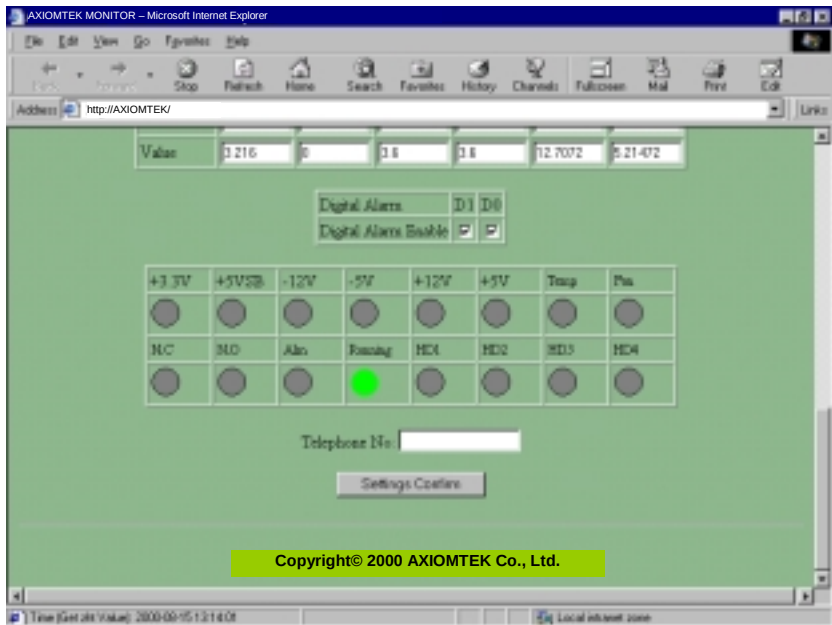
Through the following user interfaces, you can view the speed of fan and control them.



By following user interface, you can view temperature/voltage/digital, and control them. When alarm occurs, the indicator light turns to red.



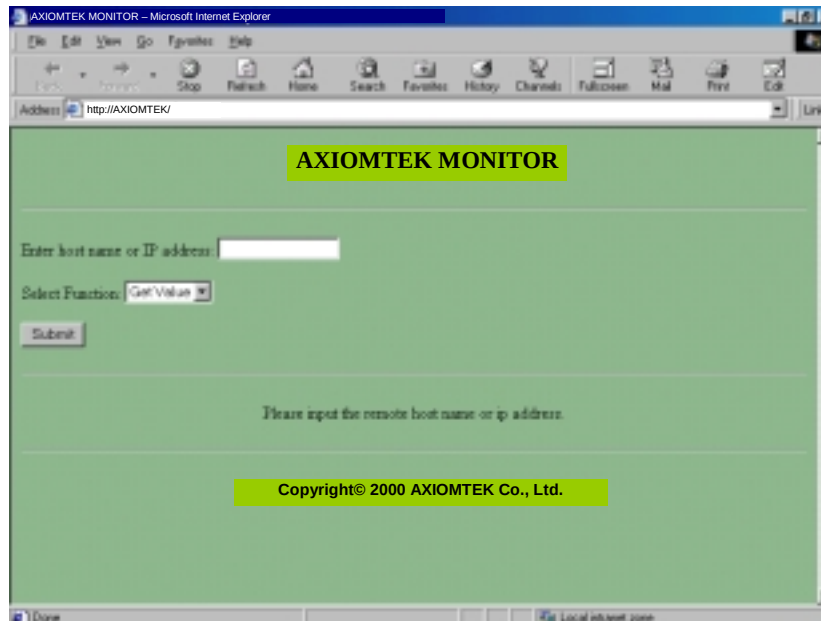
If you want to control the remote host, click the button named “Settings confirm”, then settings will be sent to host.



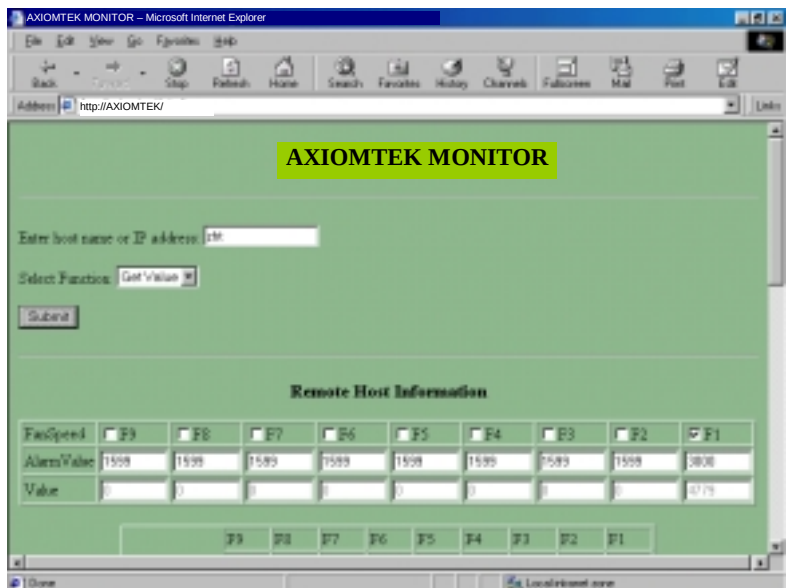
4.2 Default.asp

Before starting, you must install and register the AXIOMTEK monitoring ActiveX control. Usually it has been done by setup program. Note that web server; IIS or PWS is necessary. This homepage is just a demo showing how to use ASP to control your AX8905.

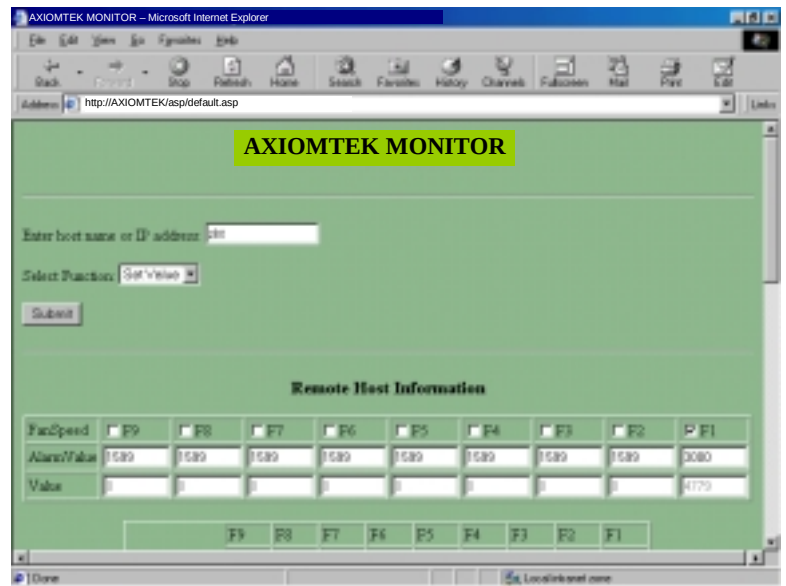
The default.asp is shown as follows:



After entering host name or IP address, select the “Get Value” option and click the button named “Submit”, then you can get remote host information. It is displayed on the screen as follows:



If you want to control the remote host, select the “Set Value” option and click the button named “Submit”, then settings will be sent to host.



4.3 Index.html and AX8905.cgi

Before starting, you must copy index.html and AX8905.cgi to proper directories. Note that web server; Apache is necessary. This homepage is just a demo showing how to use CGI program to control your AX8905.

Its display on screen is similar to default.asp. So we do not repeat its description in this section.

This page does not contain any information.

Chapter 5

Function List

In this chapter, we provide the brief description of each function supported. You may use these functions to re-design your homepage.

5.1 Windows

5.1.1 ConnectRemoteHost

■ **Description**

This is an initial function must be called before calling all functions. Call this function to establish a connection to remote host.

■ **Syntax**

BOOL ConnectRemoteHost(BSTR bstrHostAddr);

■ **Argument**

● **bstrHostAddr**

Remote host name or IP address.

■ **Return**

TRUE : Connect OK.

FALSE : Connect failure.

5.1.2 GetVoltEnable

■ **Description**

Use this function to see if the voltage-monitoring item of remote host is disabled or enabled.

■ **Syntax**

short GetVoltEnable(BSTR bstrHostAddr, short nVoltIndex, VARIANT *pnVoltEnable)

■ **Argument**

● **bstrHostAddr**

Remote host name or IP address.

● **nVoltIndex**

The index of voltage

● ***pnVoltEnable**

1 : the voltage-monitoring item of remote host is enabled.

2 : the voltage-monitoring item of remote host is disabled.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.3 GetVoltAlarm

■ **Description**

Use this function to know that the voltage alarm of remote host is generating or not.

■ **Syntax**

short GetVoltAlarm(BSTR bstrHostAddr, short nVoltIndex, VARIANT *pnVoltAlarm)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nVoltIndex**
The index of voltage
- ***pnVoltAlarm**
 - 1 : the voltage alarm of remote host is generating
 - 0 : the voltage alarm of remote host is not generating

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.4 GetVoltSetting

■ **Description**

Use this function to know the voltage setting value of remote host.

■ **Syntax**

short GetVoltSetting(BSTR bstrHostAddr, short nVoltIndex, VARIANT *pfltVoltSetting)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nVoltIndex**
The index of voltage
- ***pfltVoltSetting**
The voltage setting value of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.5 GetVoltReading

■ **Description**

Use this function to know the voltage reading value of remote host.

■ **Syntax**

short GetVoltReading(BSTR bstrHostAddr, short nVoltIndex, VARIANT *pfltVoltReading)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nVoltIndex**
The index of voltage
- ***pfltVoltReading**
The voltage reading value of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.6 GetTempEnable

■ **Description**

Use this function to see if the temperature-monitoring item of remote host is disabled or enabled.

■ **Syntax**

short GetTempEnable(BSTR bstrHostAddr, short nTempIndex, VARIANT *pnTempEnable)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nTempIndex**
The index of temperature.
- ***pnTempEnable**
 - 1 : the temperature-monitoring item of remote host is enabled
 - 0 : the temperature-monitoring item of remote host is disabled

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.7 GetTempAlarm

■ **Description**

Use this function to know that the temperature alarm of remote host is generating or not.

■ **Syntax**

short GetTempAlarm(BSTR bstrHostAddr, short nTempIndex, VARIANT *pnTempAlarm)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nTempIndex**
The index of temperature.
- ***pnTempAlarm**
 - 1 : the temperature alarm of remote host is generating
 - 0 : the temperature alarm of remote host is not generating

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.8 GetTempSetting

■ **Description**

Use this function to know the temperature setting value of remote host.

■ **Syntax**

short GetTempSetting(BSTR bstrHostAddr, short nTempIndex, VARIANT *pfltTempSetting)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nTempIndex**
The index of temperature.
- ***pfltTempSetting**
The temperature setting value of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.9 GetTempReading

■ **Description**

Use this function to know the temperature reading value of remote host.

■ **Syntax**

short GetTempReading(BSTR bstrHostAddr, short nTempIndex, VARIANT *pfltTempReading)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nTempIndex**
The index of temperature.
- ***pfltTempReading**
The temperature reading value of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.10 GetFanEnable

■ **Description**

Use this function to know that the fan-monitoring item of remote host is disabled or enabled.

■ **Syntax**

short GetFanEnable(BSTR bstrHostAddr, short nFanIndex, VARIANT *pnFanEnable)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nFanIndex**
The index of fan.
- ***pnFanEnable**
 - 1 : the fan-monitoring item of remote host is enabled
 - 0 : the fan-monitoring item of remote host is disabled

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.11 GetFanAlarm

■ **Description**

Use this function to know that the fan alarm of remote host is generating or not.

■ **Syntax**

short GetFanAlarm(BSTR bstrHostAddr, short nFanIndex, VARIANT *pnFanAlarm)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nFanIndex**
The index of fan.
- ***pnFanAlarm**
 - 1 : the fan alarm of remote host is generating
 - 0 : the fan alarm of remote host is not generating

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.12 GetFanSetting

■ **Description**

Use this function to know the fan speed setting value of remote host.

■ **Syntax**

short GetFanSetting(BSTR bstrHostAddr, short nFanIndex, VARIANT *pfltFanSetting)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nFanIndex**
The index of fan.
- ***pfltFanSetting**
The fan speed setting value of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.13 GetFanReading

■ **Description**

Use this function to know the fan speed reading value of remote host.

■ **Syntax**

short GetFanReading(BSTR bstrHostAddr, short nFanIndex, VARIANT *pfltFanReading)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nFanIndex**
The index of fan
- ***pfltFanReading**
The fan speed reading value of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.14 GetDigitalEnable

■ **Description**

Use this function to see if the digital alarm-monitoring item of remote host is disabled or enabled.

■ **Syntax**

short GetDigitalEnable(BSTR bstrHostAddr, short nDigitalIndex, VARIANT *pnDigitalEnable)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nDigitalIndex**
The index of digital alarm.
- ***pnDigitalEnable**
 - 1 : the digital alarm-monitoring item of remote host is enabled
 - 0 : the digital alarm-monitoring item of remote host is disabled

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.15 GetDigitalAlarm

■ **Description**

Use this function to know that the digital alarm of remote host is generating or not.

■ **Syntax**

short GetDigitalAlarm(BSTR bstrHostAddr, short nDigitalIndex, VARIANT *pnDigitalAlarm)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nDigitalIndex**
The index of digital alarm.
- ***pnDigitalAlarm**
 - 1 : the digital alarm of remote host is generating
 - 0 : the digital alarm of remote host is not generating

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.16 GetTrimFanSpeed

■ **Description**

Use this function to get the trimmed fan speed value of remote host.

■ **Syntax**

short GetTrimFanSpeed(BSTR bstrHostAddr, short nFanIndex, VARIANT *pnTrimFanSpeed)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nFanIndex**
The index of fan.
- ***pnTrimFanSpeed**
The trimmed fan speed value of remote host.(0..255)

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.17 GetTelNo

■ **Description**

Use this function to get the telephone number of remote host.

■ **Syntax**

short GetTelNo(BSTR bstrHostAddr, VARIANT *pbstrTelNo)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- ***pbstrTelNo**
The telephone number of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed

5.1.18 SetVoltEnable

■ **Description**

Use this function to disable or enable the voltage-monitoring item of remote host.

■ **Syntax**

short SetVoltEnable(BSTR bstrHostAddr, short nVoltIndex, short nVoltEnable)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nVoltIndex**
The index of voltage
- **nVoltEnable**
 - 1 : enable the voltage-monitoring item of remote host.
 - 0 : disable the voltage-monitoring item of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.19 SetVoltSetting

■ **Description**

Use this function to set the voltage setting value of remote host.

■ **Syntax**

short SetVoltSetting(BSTR bstrHostAddr, short nVoltIndex, float fltVoltSetting)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nVoltIndex**
The index of voltage
- **fltVoltSetting**
The voltage setting value is sent to remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.20 SetTempEnable

■ **Description**

Use this function to disable or enable the temperature-monitoring item of remote host.

■ **Syntax**

short SetTempEnable(BSTR bstrHostAddr, short nTempIndex, short nTempEnable)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nTempIndex**
The index of temperature.
- **nTempEnable**
 - 1 : enable the temperature-monitoring item of remote host.
 - 0 : disable the temperature-monitoring item of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.21 SetTempSetting

■ **Description**

Use this function to set the temperature setting value of remote host.

■ **Syntax**

short SetTempSetting(BSTR bstrHostAddr, short nTempIndex, float fltTempSetting)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nTempIndex**
The index of temperature
- **fltTempSetting**
The temperature setting value is sent to remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.22 SetFanEnable

■ **Description**

Use this function to disable or enable the fan-monitoring item of remote host.

■ **Syntax**

short SetFanEnable(BSTR bstrHostAddr, short nFanIndex, short nFanEnable)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nFanIndex**
The index of fan.
- **nFanEnable**
 - 1 : enable the fan-monitoring item of remote host.
 - 0 : disable the fan-monitoring item of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.23 SetFanSetting

■ **Description**

Use this function to set the fan speed setting value of remote host.

■ **Syntax**

short SetFanSetting(BSTR bstrHostAddr, short nFanIndex, float fltFanSetting)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nFanIndex**
The index of fan
- **fltFanSetting**
The fan speed setting value is sent to remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.24 SetDigitalEnable

■ **Description**

Use this function to disable or enable the digital monitoring item of remote host.

■ **Syntax**

short SetDigitalEnable(BSTR bstrHostAddr, short nDigitalIndex, short nDigitalEnable)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nDigitalIndex**
The index of digital.
- **nFanEnable**
 - 1 : enable the digital monitoring item of remote host.
 - 0 : disable the digital monitoring item of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.25 SetTrimFanSpeed

■ **Description**

Use this function to trim the fan speed of remote host.

■ **Syntax**

short SetTrimFanSpeed(BSTR bstrHostAddr, short nFanIndex, short nTrimFanSpeed)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **nFanIndex**
The index of fan
- **nTrimFanSpeed**
Use this value (0..255) to trim the fan speed value of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : index is incorrect

5.1.26 SetTelNo

■ **Description**

Use this function to set the telephone number of remote host.

■ **Syntax**

short SetTelNo(BSTR bstrHostAddr, BSTR bstrTelNo)

■ **Argument**

- **bstrHostAddr**
Remote host name or IP address.
- **bstrTelNo**
Use this string to set the telephone number of remote host.

■ **Return**

- 0 : success
- 1 : socket is closed
- 2 : the size of bstrTelNo is incorrect

5.1.27 SendAllValueToRemoteHost

■ **Description**

This function must be called after calling other “Set*” functions.

■ **Syntax**

short SendAllValueToRemoteHost(BSTR bstrHostAddr)

■ **Argument**

● **bstrHostAddr**

Remote host name or IP address.

■ **Return**

0 : send OK.

-1 : socket is closed.

5.2 Linux

5.2.1 GetRemoteHostData

■ **Description**

This is an initial function must be called before calling all functions. Call this function to establish a connection to remote host and get data.

■ **Syntax**

int GetRemoteHostData(char * strHostName)

■ **Argument**

● **strHostName**

Remote host name or IP address.

■ **Return**

1 : Connect and get data OK.

0 : Connect or get data failure.

5.2.2 GetVoltEnable

■ **Description**

Use this function to see if the voltage-monitoring item of remote host is disabled or enabled.

■ **Syntax**

short GetVoltEnable(short nVoltIndex, char * pbVoltEnable)

■ **Argument**

● **nVoltIndex**

The index of voltage

- ***pbVoltEnable**
 - 1 : the voltage-monitoring item of remote host is enabled.
 - 0 : the voltage-monitoring item of remote host is disabled.

■ **Return**

- 0 : success
- 2 : index is incorrect

5.2.3 GetVoltAlarm

■ **Description**

Use this function to know that the voltage alarm of remote host is generating or not.

■ **Syntax**

short GetVoltAlarm(short nVoltIndex, char * pbVoltAlarm)

■ **Argument**

- **nVoltIndex**
 - The index of voltage
- ***pbVoltAlarm**
 - 1 : the voltage alarm of remote host is generating
 - 0 : the voltage alarm of remote host is not generating

■ **Return**

- 0 : success
- 2 : index is incorrect

5.2.4 GetVoltSetting

■ **Description**

Use this function to know the voltage setting value of remote host.

■ **Syntax**

short GetVoltSetting(short nVoltIndex, float * pfltVoltSetting)

■ **Argument**

- **nVoltIndex**
 - The index of voltage
- ***pfltVoltSetting**
 - The voltage setting value of remote host.

■ **Return**

- 0 : success
- 2 : index is incorrect

5.2.5 GetVoltReading

■ **Description**

Use this function to know the voltage reading value of remote host.

■ **Syntax**

short GetVoltReading(short nVoltIndex, float * pfltVoltReading)

■ **Argument**

● **nVoltIndex**

The index of voltage

● ***pfltVoltReading**

The voltage reading value of remote host.

■ **Return**

0 : success

-2 : index is incorrect

5.2.6 GetTempEnable

■ **Description**

Use this function to know that the temperature-monitoring item of remote host is disabled or enabled.

■ **Syntax**

short GetTempEnable(short nTempIndex, char * pbTempEnable)

■ **Argument**

● **nTempIndex**

The index of temperature.

● ***pbTempEnable**

1 : the temperature-monitoring item of remote host is enabled

0 the temperature monitoring item of remote host is disabled

■ **Return**

0 : success

-2 : index is incorrect

5.2.7 GetTempAlarm

■ **Description**

Use this function to know that the temperature alarm of remote host is generating or not.

■ **Syntax**

short GetTempAlarm(short nTempIndex, char * pbTempAlarm)

■ **Argument**

● **nTempIndex**

The index of temperature.

● ***pbTempAlarm**

1 : the temperature alarm of remote host is generating

0 : the temperature alarm of remote host is not generating

■ **Return**

0 : success

-2 : index is incorrect

5.2.8 GetTempSetting

■ **Description**

Use this function to know the temperature setting value of remote host.

■ **Syntax**

short GetTempSetting(short nTempIndex, char * pcTempSetting)

■ **Argument**

● **nTempIndex**

The index of temperature.

● ***pcTempSetting**

The temperature setting value of remote host.

■ **Return**

0 : success

-2 : index is incorrect

5.2.9 GetTempReading

■ **Description**

Use this function to know the temperature reading value of remote host.

■ **Syntax**

short GetTempReading(short nTempIndex, char * pcTempReading)

■ **Argument**

● **nTempIndex**

The index of temperature.

● ***pcTempReading**

The temperature reading value of remote host.

■ **Return**

0 : success

-2 : index is incorrect

5.2.10 GetFanEnable

■ **Description**

Use this function to know that the fan-monitoring item of remote host is disabled or enabled.

■ **Syntax**

short GetFanEnable(short nFanIndex, char * pbFanEnable)

■ **Argument**

● **nFanIndex**

The index of fan.

● ***pbFanEnable**

1 : the fan-monitoring item of remote host is enabled

0 : the fan-monitoring item of remote host is disabled

■ **Return**

0 : success

-2 : index is incorrect

5.2.11 GetFanAlarm

■ **Description**

Use this function to know that the fan alarm of remote host is generating or not.

■ **Syntax**

short GetFanAlarm(short nFanIndex, char * pbFanAlarm)

■ **Argument**

● **nFanIndex**

The index of fan.

● ***pbFanAlarm**

1 : the fan alarm of remote host is generating

0 : the fan alarm of remote host is not generating

■ **Return**

0 : success

-2 : index is incorrect

5.2.12 GetFanSetting

■ **Description**

Use this function to know the fan speed setting value of remote host.

■ **Syntax**

short GetFanSetting(short nFanIndex, unsigned short * pwFanSetting)

■ **Argument**

● **nFanIndex**

The index of fan.

● ***pwFanSetting**

The fan speed setting value of remote host.

■ **Return**

0 : success

-2 : index is incorrect

5.2.13 GetFanReading

■ **Description**

Use this function to know the fan speed reading value of remote host.

■ **Syntax**

short GetFanReading(short nFanIndex, unsigned short * pwFanReading)

■ **Argument**

● **nFanIndex**

The index of fan

● ***pwFanReading**

The fan speed reading value of remote host..

■ **Return**

0 : success

-2 : index is incorrect

5.2.14 GetDigitalEnable

■ **Description**

Use this function to see if the digital alarm-monitoring item of remote host is disabled or enabled.

■ **Syntax**

short GetDigitalEnable(short nDigitalIndex, char * pbDigitalEnable)

■ **Argument**

● **nDigitalIndex**

The index of digital alarm.

● ***pbDigitalEnable**

1 : the digital alarm-monitoring item of remote host is enabled

0 : the digital alarm-monitoring item of remote host is disabled

■ **Return**

0 : success

-2 : index is incorrect

5.2.15 GetDigitalAlarm

■ **Description**

Use this function to know that the digital alarm of remote host is generating or not.

■ **Syntax**

short GetDigitalAlarm(short nDigitalIndex, char * pbDigitalAlarm)

■ **Argument**

● **nDigitalIndex**

The index of digital alarm.

● ***pbDigitalAlarm**

1 : the digital alarm of remote host is generating

0 : the digital alarm of remote host is not generating

■ **Return**

0 : success

-2 : index is incorrect

5.2.16 GetTrimFanSpeed

■ **Description**

Use this function to get the trimmed fan speed value of remote host.

■ **Syntax**

short GetTrimFanSpeed(short nFanIndex, unsigned char * pbyteTrimFanSpeed)

■ **Argument**

● **nFanIndex**

The index of fan.

● ***pbyteTrimFanSpeed**

The trimmed fan speed value of remote host.(0..255)

■ **Return**

0 : success

-2 : index is incorrect

5.2.17 GetTelNo

■ **Description**

Use this function to get the telephone number of remote host.

■ **Syntax**

short GetTelNo(char * strTelNo)

■ **Argument**

- **strTelNo**

The telephone number of remote host.

■ **Return**

0 : success

5.2.18 SetVoltEnable

■ **Description**

Use this function to disable or enable the voltage-monitoring item of remote host.

■ **Syntax**

short SetVoltEnable(short nVoltIndex, char bVoltEnable)

■ **Argument**

- **nVoltIndex**

The index of voltage

- **bVoltEnable**

1 : enable the voltage-monitoring item of remote host.

0 : disable the voltage-monitoring item of remote host.

■ **Return**

0 : success

-2 : index is incorrect

5.2.19 SetVoltSetting

■ **Description**

Use this function to set the voltage setting value of remote host.

■ **Syntax**

short SetVoltSetting(short nVoltIndex, float * pfltVoltSetting)

■ **Argument**

- **nVoltIndex**

The index of voltage

- ***pfltVoltSetting**

The voltage setting value is sent to remote host.

■ **Return**

- 0 : success
- 2 : index is incorrect

5.2.20 SetTempEnable

■ **Description**

Use this function to disable or enable the temperature-monitoring item of remote host.

■ **Syntax**

short SetTempEnable(short nTempIndex, char bTempEnable)

■ **Argument**

- **nTempIndex**

The index of temperature.

- **bTempEnable**

- 1 : enable the temperature-monitoring item of remote host.
- 0 : disable the temperature-monitoring item of remote host.

■ **Return**

- 0 : success
- 2 : index is incorrect

5.2.21 SetTempSetting

■ **Description**

Use this function to set the temperature setting value of remote host.

■ **Syntax**

short SetTempSetting(short nTempIndex, char cTempSetting)

■ **Argument**

- **cTempIndex**

The index of temperature

- **cTempSetting**

The temperature setting value is sent to remote host.

■ **Return**

- 0 : success
- 2 : index is incorrect

5.2.22 SetFanEnable

■ **Description**

Use this function to disable or enable the fan-monitoring item of remote host.

■ **Syntax**

short SetFanEnable(short nFanIndex, char bFanEnable)

■ **Argument**

● **nFanIndex**

The index of fan.

● **bFanEnable**

1 : enable the fan-monitoring item of remote host.

0 : disable the fan-monitoring item of remote host.

■ **Return**

0 : success

-2 : index is incorrect

5.2.23 SetFanSetting

■ **Description**

Use this function to set the fan speed setting value of remote host.

■ **Syntax**

short SetFanSetting(short nFanIndex, unsigned short wFanSetting)

■ **Argument**

● **nFanIndex**

The index of fan

● **wFanSetting**

The fan speed setting value is sent to remote host.

■ **Return**

0 : success

-2 : index is incorrect

5.2.24 SetDigitalEnable

■ **Description**

Use this function to disable or enable the digital monitoring item of remote host.

■ **Syntax**

short SetDigitalEnable(short nDigitalIndex, char bDigitalEnable)

■ **Argument**

● **nDigitalIndex**

The index of digital.

● **bDigitalEnable**

1 : enable the digital monitoring item of remote host.

0 : disable the digital monitoring item of remote host.

■ **Return**

0 : success

-2 : index is incorrect

5.2.25 SetTrimFanSpeed

■ **Description**

Use this function to trim the fan speed of remote host.

■ **Syntax**

short SetTrimFanSpeed(short nFanIndex, unsigned char byteTrimFanSpeed)

■ **Argument**

● **nFanIndex**

The index of fan

● **byteTrimFanSpeed**

Use this value (0..255) to trim the fan speed value of remote host.

■ **Return**

0 : success

-2 : index is incorrect

5.2.26 SetTelNo

■ **Description**

Use this function to set the telephone number of remote host.

■ **Syntax**

short SetTelNo(char * strTelNo)

■ **Argument**

● **strTelNo**

Use this string to set the telephone number of remote host.

■ **Return**

0 : success

-2 : the size of bstrTelNo is incorrect

5.2.27 SendDataToRemoteHost

■ **Description**

This function must be called after calling other "Set*" functions.

■ **Syntax**

int SendDataToRemoteHost(char * strHostName)

■ **Argument**

● **strHostName**

Remote host name or IP address.

■ **Return**

1 : send OK.

0 : send FAILURE.